ABSTRACT

A tree-based datastore comprising a forest of interconnected trees is generated and/or accessed. The tree-based datastore comprises a first tree that depends from a first root node and may include a plurality of branches. Each of the branches of the first tree ends in a leaf node. Each leaf node may represent an end product, or a subcomponent node, ~~as described more fully below.~~ A second root of the same tree-based datastore is linked to each leaf node representing an end product. ~~Hence, the second root is essentially a root to an inverted order of the first tree, but the first tree is not duplicated.~~ Finally, the tree-based datastore comprises a plurality of trees in which the root node of each of these trees ~~is~~ can be described as an elemental node, ~~as described more fully below~~. The root node of each of these trees may be linked to one or more nodes in one or more branches of the first tree. The nodes of the tree-based datastore contain only pointers to other nodes in the tree-based datastore, and may contain additional fields wherein one such may be a count field. ~~The roots of the trees in the forest of trees comprising the tree-based datastore may be linked to data~~ Additionally, means to get probabilities of the coincidence of variables related to particular nodes as identified by desired contexts within one or more defined foci are described. Further, the application of logical operators to queries regarding such variables is shown.